AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A machine-processing method for computing a property of a mathematically modelled physical system, the steps comprising:

a) reading, via a machine processing unit, input data including a value for each identified ordered coefficient of a first polynomial p(x) representing said property, said polynomial p(x) being expressed asp(x) = $\Sigma(Pj \bullet x^j)$ where j=0 to n, a value of a quantity x, a value of a predetermined x_i, and a value of a predetermined $p(x_i)$ correspondingly paired with said predetermined x_i;

b) building, via said machine processing unit, a value of a second polynomial c(x) having ordered coefficients, said second polynomial c(x) being expressible as: $c(x) = \Sigma(C_k \bullet x^k)$ where k=0 to (n-1) so that said first polynomial p(x) is expressible as: $p(x)=p(x_i)+\{x- x_i \} \bullet c(x)$, comprising the steps of:

i) determining, via said machine processing unit, a value for each ordered coefficient of said second polynomial c(x) by generating a plurality of machine processing unit signals to determine each said ordered coefficient of said second polynomial c(x) from: $C_k = \Sigma(P_{(k+1+j)} \bullet x^j_i)$ where j=0 to (n-1-k);

ii) determining, via said machine processing unit, a value of said second polynomial c(x) by generating a plurality of machine processing unit signals to determine: $c(x) = \Sigma(C_k \bullet x^k)$ where k=0 to (n-1);

c) constructing, via said machine processing unit, a value of said first polynomial p(x) by generating a plurality of machine processing unit signals to determine: $p(x)=p(x_i)+\{x-x_i\} \bullet c(x)$; and

d) outputting, via said machine-processing unit, said value of the first polynomial p(x) representing said property of the mathematically modelled physical system, wherein

said value of the first polynomial is outputted as a floating point number and the floating point number is a digital representation of an arbitrary real number in said machine processing unit.

2. (Original) The machine-implementable method of claim 1, wherein a difference between x and x_i is sufficiently small to achieve a desired accuracy of a final computation of said machine representation of a numerical value of said first polynomial p(x).

3. (Original) The machine-implementable method of claim 2 wherein the step of reading said input data comprises reading, via said machine processing unit, said input data from a machine-readable medium.

4. (Original) The machine-implementable method of claim 3 wherein said ordered coefficients of said second polynomial c(x) are computed from a mathematical expression derivable from: $C_k = \Sigma(P_{(k+1+j)} \bullet x^j_i)$ where j=0 to (n-1-k).

5. (Original) The machine-implementable method of claim 4 wherein said mathematical expression is a mathematical recurrence expression.

6. (Original) The machine-implementable method of claim 5 wherein said mathematical recurrence expression is a forward mathematical recurrence expression.

7. (Original) The machine-implementable method of claim 5 wherein said mathematical recurrence expression is a backward mathematical recurrence expression.

8. (Original) The machine-implementable method of claim 7 further adapted to implement said backward mathematical recurrence expression by comprising further steps for:

e) equating, via said machine-processing unit, a value of a highest ordered coefficient of said second polynomial c(x) to a value of an identified highest ordered coefficient of said first polynomial p(x) by generating a plurality of machine processing unit signals to determine: $C_{n-1}=P_n$; and

f) computing, via a machine processing unit, a value for each lower ordered coefficient of said second polynomial c(x) by generating a plurality of machine processing unit signals to determine:C_k-1=$x_i \bullet C_k+P_k$ where k = (n-1) to 1.

9. (Original) The machine-implementable method of claim 8 wherein said predetermined x_i is selected from a set of predetermined values of x_i.

10. (Original) The machine-implementable method of claim 9 wherein said predetermined x_i is a closest member of said set to said identified x.

11. (Original) The machine-implementable method of claim 10 wherein said step of determining a value of said second polynomial c(x) is computed by using Homer's Rule.

12. (Original) The machine-implementable method of claim 11 for determining a value of a denominator polynomial q(x) having identified ordered denominator coefficients, said denominator polynomial q(x) being expressible as: $q(x) = \Sigma(Q_j \bullet x^j)$ where j=0 to m, comprising further steps of:

g) adapting said input data to further include a value for each identified ordered denominator coefficient of said denominator polynomial q(x), a value of a predetermined $q(x_i)$ correspondingly paired with said predetermined x_i; and

h) determining, via said machine processing unit, a value of another polynomial d(x) having ordered denominator coefficients, said another polynomial d(x) being expressible as: $d(x) = \Sigma(D_k \bullet x^k)$ where k = 0 to (m-1) so that said denominator polynomial q(x) is expressible as: $q(x) = q(x_i)+\{x- x_i \} \bullet d(x)$, and a value for the said denominator polynomial is resolved.

13. (Original) The machine-implementable method of claim 12 wherein the first polynomial p(x) is a numerator polynomial p(x), and $p(x)-p(x_i)$ is computed, and $p(x_i)$ is not read.

14. (Original) The machine-implementable method of claim 13 for determining a value of a rational function r(x) being expressible as a quotient of said numerator polynomial p(x) and said denominator polynomial q(x) expressed as r(x) = p(x) / q(x), comprising further steps of:

i) adapting said input data to further including a value of a predetermined $r(x_i)$ correspondingly paired with said predetermined x_i; and

j) constructing, via said machine processing unit, a value of said rational function r(x) by generating a plurality of machine processing unit signals to determine:

$$r(x) = r(x_i) \bullet (1 - (q(x)-q(x_i))/q(x))) + (p(x) - p(x_i))/q(x) .$$

15. (Original) The machine-implementable method of claim 14 wherein said rational function r(x) is an approximation to a Bessel function.

16. (Original) The machine-implementable method of claim 14 wherein said rational function r(x) is an approximation to an error function (ERF).

17. (Original) The machine-implementable method of claim 14 wherein said rational function r(x) is an approximation to a complementary error function (ERFC).

18. (Original) The machine-implementable method of claim 14 wherein said rational function r(x) is an approximation to a log gamma function (LGAMMA).

19. (Original) The machine-implementable method of claim 11 or 14 wherein all values are machine representations of some numerical value, said machine processing unit is a computer processing unit, each machine representation is a binary representation operable with said computer processing unit, and machine-readable medium is a computer-readable medium.

Claims 20-22 (Cancelled)

23. (Currently Amended) A machine for computing a property of a mathematically modelled physical system, the machine configured to perform the steps comprising:

a) reading, via a machine processing unit, input data including a value for each identified ordered coefficient of a first polynomial p(x) representing said property, said polynomial p(x) being expressed as $p(x) = \Sigma(P_j \bullet x^j)$ where j = 0 to n, a value of a quantity x, a value of a predetermined x_i, and a value of a predetermined $p(x_i)$ correspondingly paired with said predetermined x_i;

b) building, via said machine processing unit, a value of a second polynomial c(x) having ordered coefficients, said second polynomial c(x) being expressible as: $c(x) = \Sigma(C_k \bullet x^k)$ where k = 0 to (n-1) so that said first polynomial p(x) is expressible as: $p(x) = p(x_i)+\{x-x_i\} \bullet c(x)$, comprising the steps of:

i) determining, via said machine processing unit, a value for each ordered coefficient of said second polynomial c(x) by generating a plurality of machine processing unit signals to determine each said ordered coefficient of said second polynomial c(x) from: $C_k=\Sigma(P_{(k+1+j)}\ x^j_i)$ where j= 0 to (n-1-k);

ii) determining, via said machine processing unit, a value of said second polynomial c(x) by generating a plurality of machine processing unit signals to determine: $c(x) = \Sigma(C_k \bullet x^k)$ where $k = 0$ to $(n-1)$;

c) constructing, via said machine processing unit, a value of said first polynomial p(x) by generating a plurality of machine processing unit signals to determine: $p(x)=p(x_i)+\{x- x_i\} \bullet c(x)$; and

d) outputting, via said machine-processing unit, said value of the first polynomial p(x) representing said property of the mathematically modelled physical system, wherein said value of the first polynomial is outputted as a floating point number and the floating point number is a digital representation of an arbitrary real number in said machine for computing.

24. (Original) The machine of claim 23 wherein a difference between x and x_i is sufficiently small to achieve a desired accuracy of a final computation of said machine representation of a numerical value of said first polynomial p(x).

25. (Original) The machine of claim 24 wherein said means for reading said input data comprises means for reading, via said machine processing unit, said input data from a machine-readable medium.

26. (Original) The machine of claim 25 wherein said ordered coefficients of said second polynomial c(x) are computed from a mathematical expression derivable from: $C_k = \Sigma(P_{(k+1+j)} \bullet x^j_i)$ where j= 0 to (n-1-k).

27. (Original) The machine of claim 26 wherein said mathematical expression is a mathematical recurrence expression.

28. (Original) The machine of claim 27 wherein said mathematical recurrence expression is a forward mathematical recurrence expression.

29. (Original) The machine of claim 27 wherein said mathematical recurrence expression is a backward mathematical recurrence expression.

30. (Original) The machine of claim 29 further adapted to implement said backward mathematical recurrence expression by further comprising:

e) means for equating, via said machine processing unit, a value of a highest ordered coefficient of said second polynomial c(x) to a value of an identified highest ordered coefficient of said first polynomial p(x) by generating a plurality of machine processing unit signals to determine: $C_{n-1} = P_n$; and

f) means for computing, via said machine processing unit, a value for each lower ordered coefficient of said second polynomial c(x) by generating a plurality of machine processing unit signals to determine: $C_{k+1} = x_i \bullet C_k+P_k$ where k = (n-1) to 1.

31. (Original) The machine of claim 30 wherein said predetermined x_i is selected from a set of predetermined values of x_i.

32. (Original) The machine of claim 30 wherein said predetermined x_i is a closest member of said set to said identified x.

33. (Original) The machine of claim 32 wherein the determining means for determining a value of said second polynomial c(x) is computed by using Homer's Rule.

34. (Original) The machine of claim 33 for determining a value of a denominator polynomial q(x) having identified ordered denominator coefficients, said denominator polynomial q(x) being expressible as: $q(x) = \Sigma(Q_j \bullet x^j)$ where j = 0 to m, comprising further steps of:

g) adapting said input data to further include a value for each identified ordered denominator coefficient of said denominator polynomial q(x), and a value of a predetermined $q(x_i)$ correspondingly paired with said predetermined x_i; and

h) determining, via said machine processing unit, a value of another polynomial d(x) having ordered denominator coefficients, said another polynomial d(x) being expressible as: $d(x) = \Sigma(D_k \bullet x^k)$ where k = 0 to (m-1) so that said denominator polynomial q(x) is expressible as: $q(x)=q(x_i)+\{x- x_i\} \bullet d(x)$, and a value for the said denominator polynomial is resolved.

35. (Original) The machine of claim 34 wherein the first polynomial p(x) is a numerator polynomial p(x), and $p(x)-p(x_i)$ is computed, and $p(x_i)$ is not read.

36. (Original) The machine of claim 35 for determining a value of a rational function r(x) being expressible as a quotient of said numerator polynomial p(x) and said denominator polynomial q(x) expressed as r(x) = p(x) / q(x), comprising further steps of:

i) adapting said input data to further including a value of a predetermined $r(x_i)$ correspondingly paired with said predetermined x_i; and

j) constructing, via said machine processing unit, a value of said rational function r(x) by generating a plurality of machine processing unit signals to determine:

$r(x) = r(x_i) \bullet (1 - (q\,(x)-q\,(x_i))/q(x))) + (p(x) - p(x_i))/q(x)$.

37. (Original) The machine of claim 36 wherein said rational function is an approximation to a Bessel function.

38. (Original) The machine of claim 36 wherein said rational function is an approximation to an error function (ERF).

39. (Original) The machine of claim 36 wherein said rational function is an approximation to a complementary error function (ERFC).

40. (Original) The machine of claim 36 wherein said rational function is an approximation to a log gamma function (LGAMMA).

41. (Original) The machine of claim 33 or 36 wherein all values are machine representations of some numerical value, said machine processing unit is a computer processing unit, each machine representation is a binary representation operable with said computer processing unit, and said machine-readable medium is a computer-readable medium.

42. (Original) A machine having a computer-readable program product having computer executable instructions for instructing a computer to embody the machine of claim 41.

43. (Original) A machine having a computer-readable mathematical software routine library including computer executable instructions for instructing a computer to embody the machine of claim 41.

44. (Original) A machine having the computer-readable mathematical software routine library of claim 43 wherein said library is operatively associated with a software programming language.